



CORE-V™



OPENHW™

OpenHW Group

CORE-V: Open Source RISC-V Cores Commercial Adoption
Lesson Learned

Rick O'Connor rickoco@openhwgroup.org

T [@rickoco](https://twitter.com/rickoco) T [@openhwgroup](https://twitter.com/openhwgroup)

www.openhwgroup.org



OPENHW™
GROUP
— PROVEN PROCESSOR IP —

Outline

- OpenHW Group Overview
- Brief History of Open Source SW (OSS)
- OpenHW ~ 2 Years Later – Lessons Learned
 - Lesson 1 – Permissive Use
 - Lesson 2 – IP Quality
 - Lesson 3 – Roadmap & Ecosystem
- Summary



OPENHW GROUP™
— PROVEN PROCESSOR IP —

and



CORE-V™



- OpenHW Group is a not-for-profit, global organization registered in Ottawa, Canada and Brussels, Belgium
- OpenHW ecosystem is driven by members (corporate & academic) and individual contributors where HW and SW designers collaborate in developing open-source cores, related IP, tools and SW such as the CORE-V Family of open-source RISC-V processors
 - International footprint with developers in North America, Europe and Asia
 - Providing an infrastructure for hosting high quality open-source HW developments in line with industry best practices
 - Strong support from industry, academia and individual contributors worldwide



OPENHW GROUP™
— PROVEN PROCESSOR IP —



OPENHW GROUP™
— PROVEN PROCESSOR IP —

Industry Members

84+ Members & Partners



OPENHW GROUP™
— PROVEN PROCESSOR IP —

© OpenHW Group

May 2022



OPENHW GROUP™
— PROVEN PROCESSOR IP —

Academic Members

84+ Members & Partners



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA



Barcelona Supercomputing Center
Centro Nacional de Supercomputación



csem **ETH** zürich



ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
Université du Québec



中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES



中国科学院软件研究所
Institute of Software Chinese Academy of Sciences



POLYTECHNIQUE MONTRÉAL
WORLD-CLASS ENGINEERING



SIMON FRASER UNIVERSITY



TECHNION
Israel Institute of Technology



uOttawa

UC SANTA BARBARA



UNIVERSITY OF Southampton



UNIVERSITY OF TORONTO



OPENHW GROUP™
— PROVEN PROCESSOR IP —



OPENHW GROUP™
— PROVEN PROCESSOR IP —

Partner Ecosystem
84+ Members & Partners



aws **cādence**®



OpenUK ♦ **ORCRO**



Accounting, Legal, Banking



OPENHW GROUP™
— PROVEN PROCESSOR IP —

Working Groups & Task Groups

- Board of Directors approves elected Chairs of Working Groups and has final approval of working group recommendations
- Technical Working Group
 - Cores Task Group
 - Verification Task Group
 - SW Task Group
 - HW Task Group
- Marketing Working Group
 - University Outreach Task Group
- OpenHW Asia Working Group
- OpenHW Europe Working Group
- Together with internal OpenHW Group engineering staff, member company development engineers (FTEs / ACs) establish and execute OpenHW Group projects
 - 20+ active projects across CORE-V RTL, Verification, GCC / LLVM, IDE, RTOS, FPGA, SoC, etc. with more projects in the pipeline



Technical Working Group (TWG)

- Co-Chair: Jérôme Quévremont, Thales Research & Technology
- Drive the overall technical direction, development roadmap and project execution for all technology related activities within the OpenHW Group and oversee the Task Groups
 - TWG is essentially the OpenHW Group company's "R&D / Engineering Organization"
- OpenHW Group engineering release methodology is based on the Eclipse Development Process
 - All OpenHW Group Platinum / Gold / Silver members are also Solutions members of the Eclipse Foundation

THALES

ECLIPSE
FOUNDATION

Outline

- OpenHW Group Overview
- Brief History of Open Source SW (OSS)
- OpenHW ~2 Years Later – Lessons Learned
 - Lesson 1 – Permissive Use
 - Lesson 2 – IP Quality
 - Lesson 3 – Roadmap & Ecosystem
- Summary

Open Source is 20+ years old



**open source
initiative®**

opensource.org

- “Open Source” is the proper name of a campaign to promote the pre-existing concept of Free Software to business, and to certify licenses to a rule set.
- Christine Peterson ...suggested “Open Source” as a way to promote Free Software without the stigma of “free” in the English language.

Bruce Perens , Co-Founder of the Open Source Initiative
<https://perens.com/2017/09/26/on-usage-of-the-phrase-open-source/>

OSI 1st Decade Timeline Advocacy & Controversy

- 1998 Term coined as rebrand for software freedom; OSI formed
- 1999 Open Source Definition published: licenses standardized
- 2000 Most open source is a proprietary replacement
- 2001 “Linux is a cancer” – Microsoft [The Register](#)
- 2002 Rush of new licenses
- 2003 SCO sues IBM over Linux [Wiki SCO - Linux Disputes](#)
- 2004 Last of Microsoft’s “Halloween Documents”
- 2005 Unix now open source (Sun Solaris)
- 2006 Open Standards Requirement (OSR) published
- 2007 Java now open source
- 2008 Most CIOs understand open source as a benefit

Source: [Simon Phipps OSI](#)



OSI 2nd Decade Adoption & Ascendancy

- 2008 Most open source is “hidden” infrastructure
- 2011 Open source enabling web service business wave
- 2013 Open source powering cloud/container revolution
- 2015 “Microsoft ❤️ Linux”
- 2016 Windows Subsystem for Linux 1 (WSL1) announced
- 2017 Open source at the heart of most new software
- 2019 Windows Subsystem for Linux 2 (WSL2) announced
- 2020 Microsoft brings Linux GUI apps to Windows 10

Source: [Simon Phipps OSI](#)



Open Source HW Adoption Lessons

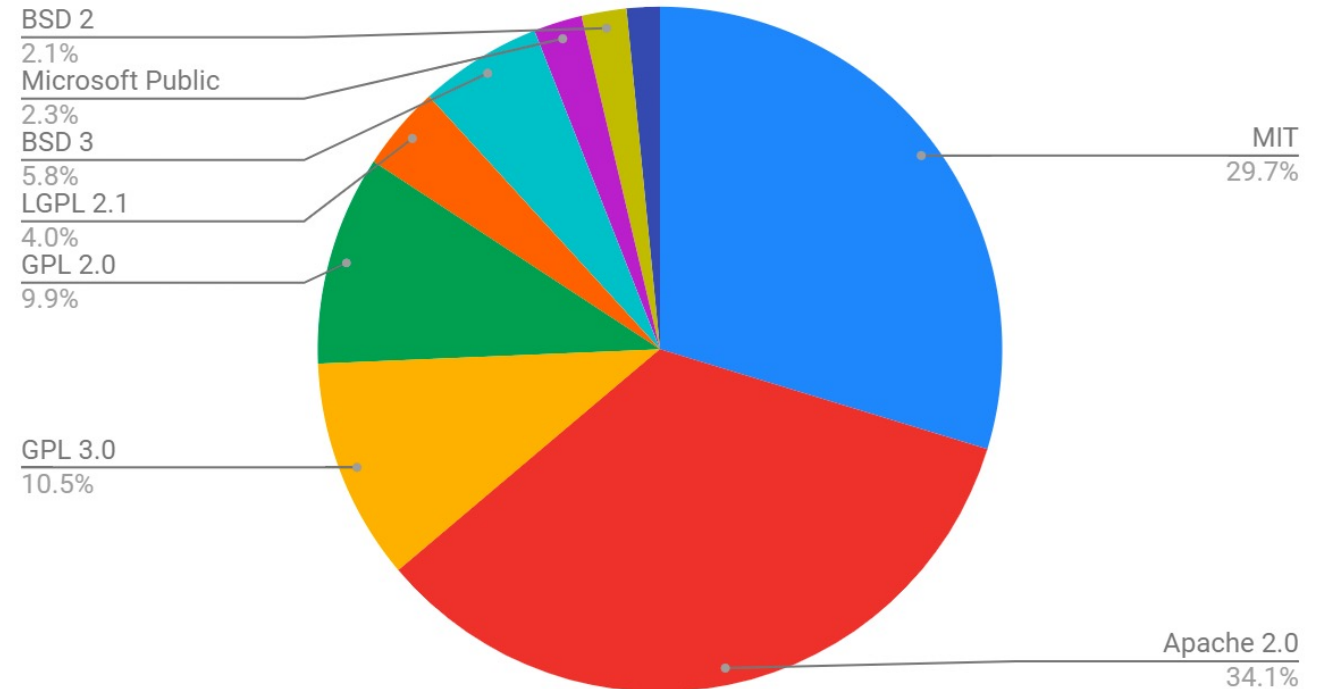
- Lesson 1 - Permissive use
 - permissive open-source licensing and processes to minimize business and legal risks
- Lesson 2 - IP quality
 - harness community best-in-class design and verification methods and contributions
- Lesson 3 - Roadmap & Ecosystem
 - ensure availability of IDE, RTOS / OS ports, physical design etc. and create a roadmap of cores covering a range of PPA metrics

Open Source License Usage - 1

- WhiteSource database > 4M open source packages and 130M open source files covering 200 programming languages
- Results show use of permissive open source licenses continues to rise, while usage of copyleft licenses, especially GPL licenses, continues to decrease.



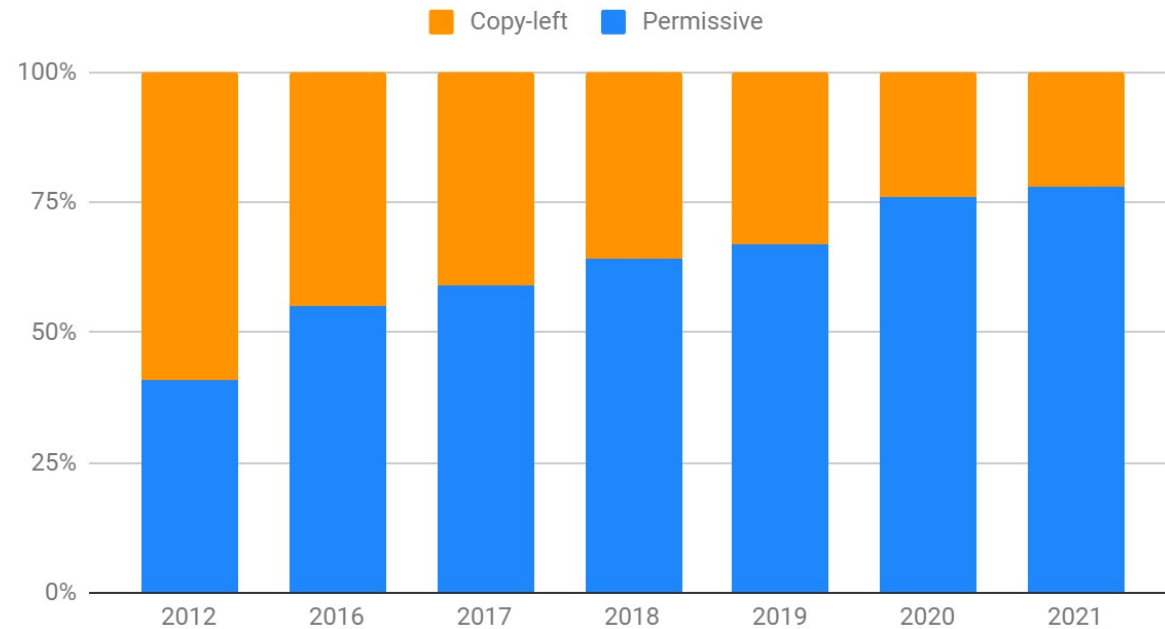
Top Open Source Licenses in 2021



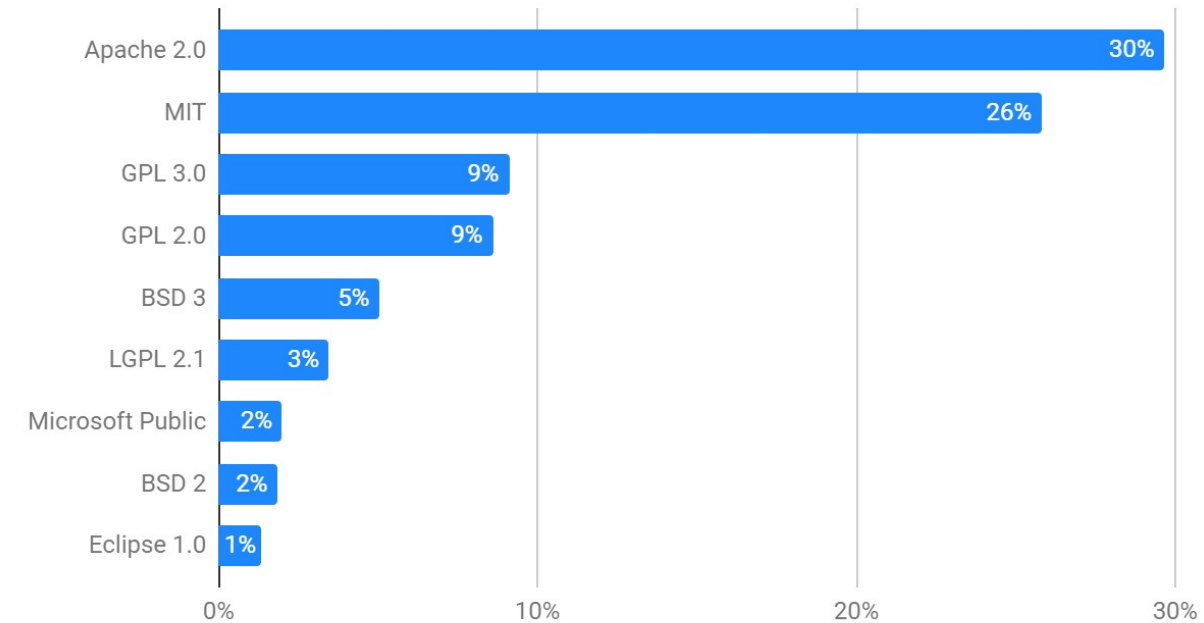
<https://www.whitesourcesoftware.com/resources/blog/open-source-licenses-trends-and-predictions/>

Open Source License Usage - 2

Permissive vs. Copy-left Licenses Over Time



Top Open Source Licenses in 2021



<https://www.whitesourcesoftware.com/resources/blog/open-source-licenses-trends-and-predictions/>

HW Companies & Open Source

- Large Systems & Semiconductor companies have very deep patent portfolios
- Copyleft / GPL style licenses are generally seen to pose a greater risk of unwanted patent exposure
- Companies need to see commercial benefit to 'give back' and not be forced to give back through license terms
- Apache provides permissive terms with both copyright and patent grants – also, most HW companies have already accepted Apache for SW projects
- But is Apache enough?



Solderpad Hardware License 2.1

- A permissive open hardware license
- Based on, and acts as an exception to, Apache-2.0
- SPDX-License-Identifier: Apache-2.0 WITH SHL-2.1
- Covers physical hardware as well as open silicon and gateware
- Modifies, clarifies and extends various Apache definitions, and the scope of rights to explicitly cover hardware
- Not specifically OSI approved, but we know it falls within the OSI definition of “open source” because any licensee can treat as plain Apache-2.0
- <http://solderpad.org/licenses/SHL-2.1/>



Solderpad: Authorship and Rights

Apache-2.0 is drafted with the terminology associated with copyright in mind (since software is primarily covered by copyright).

Copyright is extended to Rights, covering other intellectual property which can apply to hardware:

“Rights” means copyright and any similar right including design right (whether registered or unregistered), rights in semiconductor topographies (mask works) and database rights (but excluding Patents and Trademarks).

Authorship is extended to cover authorship and design.

Solderpad: Source and Object Form

- *Source* is extended to cover *net lists, board layouts, CAD files, documentation source, and configuration files.*
- *Object form* is extended to forms applicable to hardware, including physical hardware, silicon and gateware, and covers:

any form resulting from mechanical transformation or translation of a Source form or the application of a Source form to physical material, including but not limited to compiled object code, generated documentation, the instantiation of a hardware design or physical object or material and conversions to other media types, including intermediate forms such as bytecodes, FPGA bitstreams, moulds, artwork and semiconductor topographies (mask works)

Open Source HW Adoption Lessons

- Lesson 1 - Permissive use
 - permissive open-source licensing and processes to minimize business and legal risks
- Lesson 2 - IP quality
 - harness community best-in-class design and verification methods and contributions
- Lesson 3 - Roadmap & Ecosystem
 - ensure availability of IDE, RTOS / OS ports, physical design etc. and create a roadmap of cores covering a range of PPA metrics

OpenHW Verification Task Group



- Charter:
 - develop best in class verification test bench environments for the cores and IP blocks developed within the OpenHW Group
 - use the best tools for the job at hand
- Co-Chairs:
 - Robert Chu, Futurewei Technologies, Inc.
 - vacant



Industry Standard Tools

- Use industry standard languages and verification methods

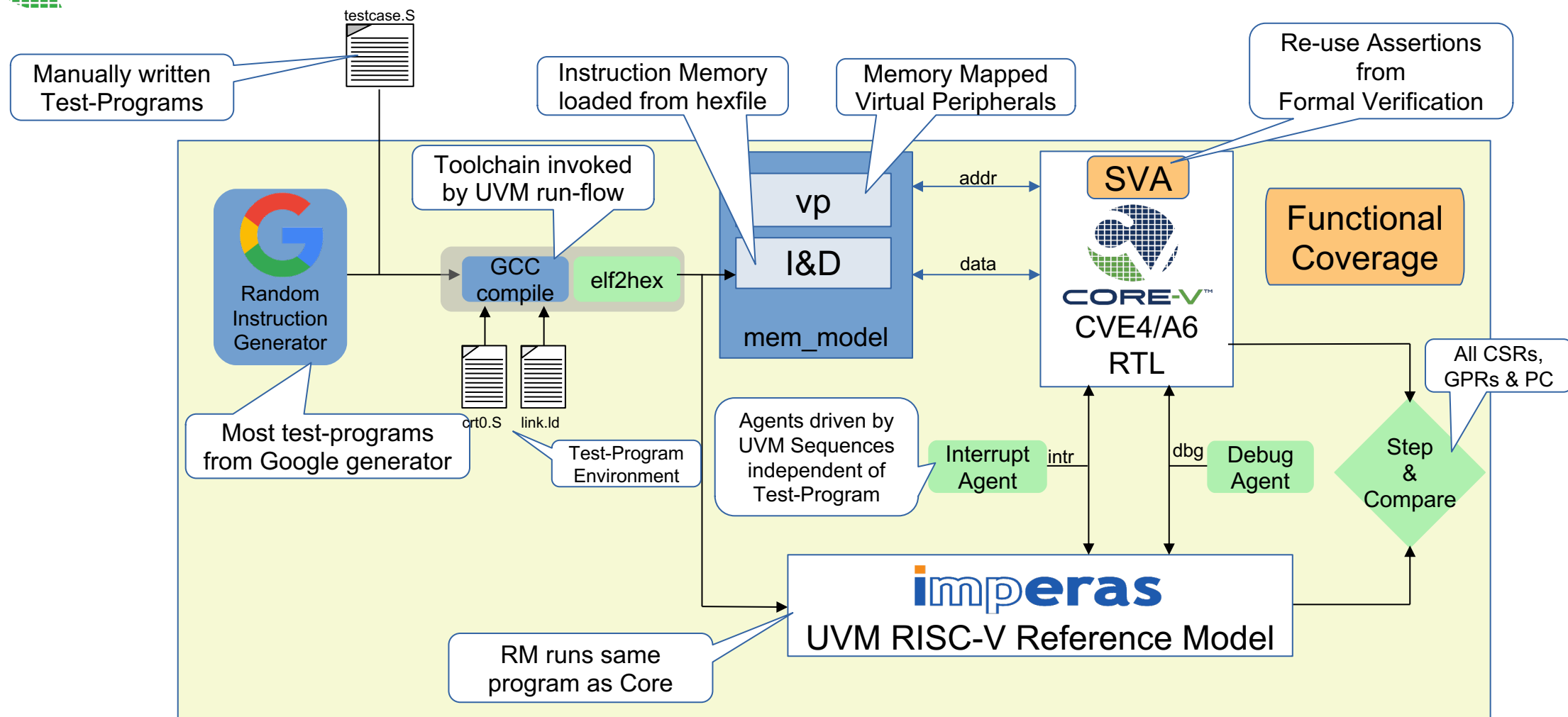


- Projects need to fit easily into ecosystem companies' EDA tool flow. Leverage the best commercial tools for the job





CORE-V™ Verif UVM Test Bench




make SIMULATOR=<sim> +UVM_TEST=riscv-dv-test



CV32E4 'RTL Freeze' Coverage Data

- CV32E4 Coverage Data published in our [core-v-verif github repo](#)
- Coverage told us we had...
 - Bugs in the functional coverage model
 - Low coverage in Instructions Exceptions testing
 - Low coverage of Int & Dbg corner-cases
 - Low coverage of Instruction and Data bus interface protocol

cadence  Metrics Report - Generated on 2020/10/23 07:16:13 [Show Details](#)

Block Expression FSM Cover Groups Assertions

Filter by name:

Name	Block Covered	Expression Covered	FSM Covered	CoverGroup Covered	Assertion Covered
cov_model	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	4758 / 4884 (97.42%)	0 / 0 (n/a)
debug_covg	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	119 / 140 (85%)	0 / 0 (n/a)
interrupt_covg	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	306 / 336 (91.07%)	0 / 0 (n/a)
isa_covg	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	4333 / 4408 (98.3%)	0 / 0 (n/a)
core_i	1242 / 1255 (98.96%)	1878 / 1930 (97.31%)	63 / 63 (100%)	0 / 0 (n/a)	264 / 265 (99.62%)
sleep_unit_i	5 / 5 (100%)	22 / 24 (91.67%)	0 / 0 (n/a)	0 / 0 (n/a)	11 / 11 (100%)
if_stage_i	159 / 159 (100%)	181 / 182 (99.45%)	14 / 14 (100%)	0 / 0 (n/a)	18 / 18 (100%)
prefetch_buffer_i	37 / 37 (100%)	82 / 83 (98.8%)	4 / 4 (100%)	0 / 0 (n/a)	15 / 15 (100%)
aligner_i	20 / 20 (100%)	38 / 38 (100%)	10 / 10 (100%)	0 / 0 (n/a)	1 / 1 (100%)
compressed_decoder_i	74 / 74 (100%)	39 / 39 (100%)	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)
genblk1	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	1 / 1 (100%)
genblk2	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	1 / 1 (100%)
id_stage_i	680 / 692 (98.27%)	651 / 664 (98.04%)	33 / 33 (100%)	0 / 0 (n/a)	17 / 18 (94.44%)
ex_stage_i	96 / 96 (100%)	679 / 682 (99.56%)	16 / 16 (100%)	0 / 0 (n/a)	4 / 4 (100%)
load_store_unit_i	101 / 101 (100%)	102 / 103 (99.03%)	0 / 0 (n/a)	0 / 0 (n/a)	5 / 5 (100%)
cs_registers_i	142 / 143 (99.3%)	160 / 191 (83.77%)	0 / 0 (n/a)	0 / 0 (n/a)	1 / 1 (100%)
genblk1	38 / 38 (100%)	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)
genblk2	41 / 42 (97.62%)	6 / 9 (66.67%)	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)
genblk3	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)
gen_trigger_regs	6 / 6 (100%)	19 / 19 (100%)	0 / 0 (n/a)	0 / 0 (n/a)	0 / 0 (n/a)

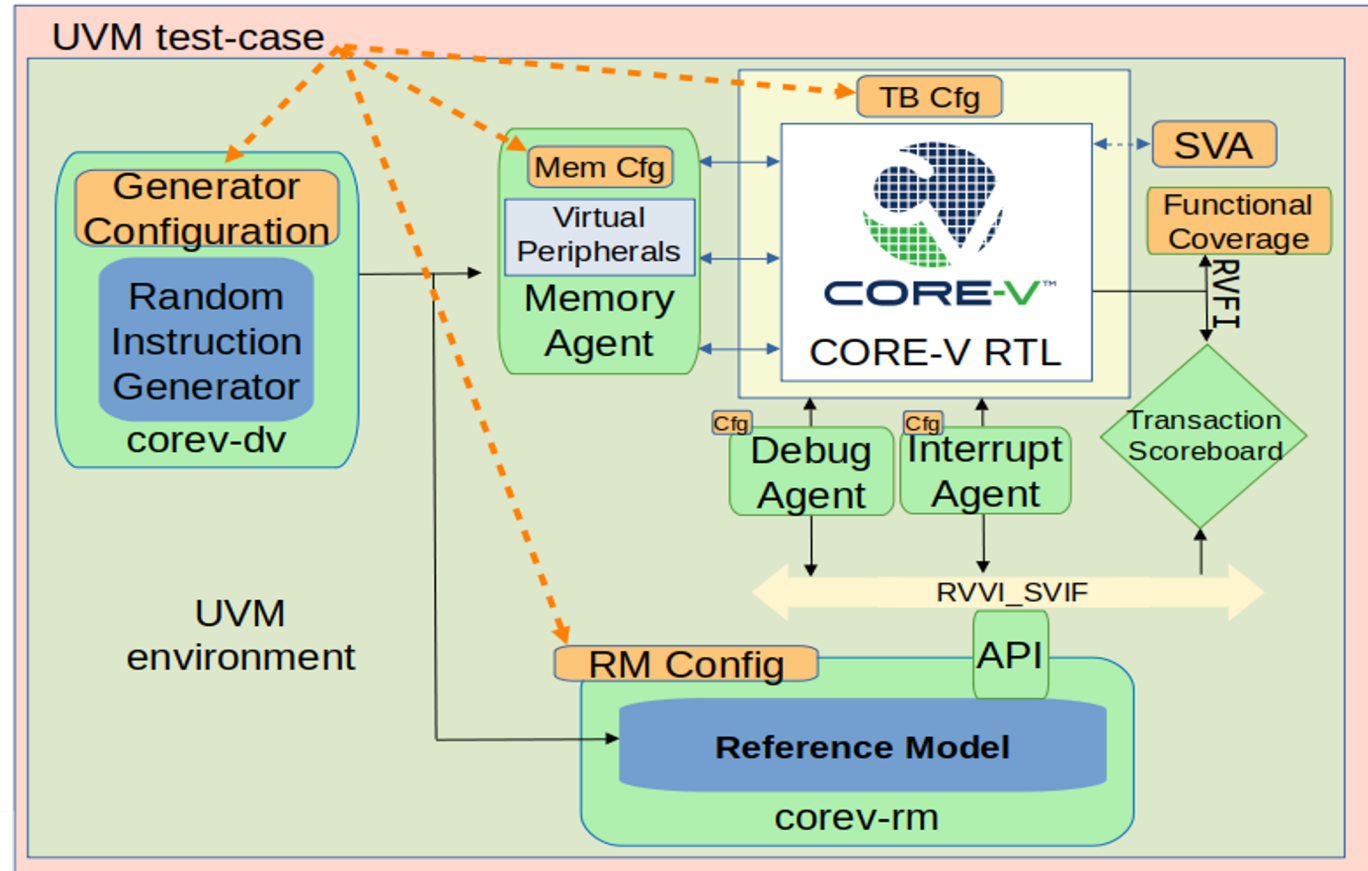
Filter by name:

Name	Overall Average Grade	Overall Covered	Enclosing Entity	Source Code
cg_debug_mode_ext	37.5%	6 / 16 (37.5%)	uvm_pkg.uvm_test_top.env.cov_model.debug_covg	covergroup cg_debug_mode_ext
cg_ebreak_execute_with_ebreakm	60%	3 / 5 (60%)	uvm_pkg.uvm_test_top.env.cov_model.debug_covg	covergroup cg_ebreak_execute_with_ebreakm
cg_cebreak_execute_with_ebreakm	100%	5 / 5 (100%)	uvm_pkg.uvm_test_top.env.cov_model.debug_covg	covergroup cg_cebreak_execute_with_ebreakm
cg_ebreak_execute_without_ebreakm	100%	3 / 3 (100%)	uvm_pkg.uvm_test_top.env.cov_model.debug_covg	covergroup cg_ebreak_execute_without_ebreakm
cg_cebreak_execute_without_ebreakm	100%	3 / 3 (100%)	uvm_pkg.uvm_test_top.env.cov_model.debug_covg	covergroup cg_cebreak_execute_without_ebreakm

Filter by name:

Overall Average Grade	Overall Covered	Source Code	Name	Overall Average Grade	Overall Covered	Score	At Least	Source Code
37.5%	6 / 16 (37.5%)	state: coverpoint cnbxt.debug_cnbxt.vif_c	auto[RESET]	0%	0 / 1 (0%)	0	1	state: coverpoint cnbxt.debug_cnbxt.vif_c
		cnbxt.debug_cnbxt.vif_mon_cb	auto[BOOT_SET]	0%	0 / 1 (0%)	0	1	state: coverpoint cnbxt.debug_cnbxt.vif_c
			auto[SLEEP]	100%	1 / 1 (100%)	6249	1	state: coverpoint cnbxt.debug_cnbxt.vif_c
			auto[WAIT_SLEEP]	100%	1 / 1 (100%)	15	1	state: coverpoint cnbxt.debug_cnbxt.vif_c
			auto[FIRST_FETCH]	0%	0 / 1 (0%)	0	1	state: coverpoint cnbxt.debug_cnbxt.vif_c
			auto[DECODE]	100%	1 / 1 (100%)	4521	1	state: coverpoint cnbxt.debug_cnbxt.vif_c
			auto[IRQ_FLUSH_ELW]	0%	0 / 1 (0%)	0	1	state: coverpoint cnbxt.debug_cnbxt.vif_c
			auto[ELW_EXE]	0%	0 / 1 (0%)	0	1	state: coverpoint cnbxt.debug_cnbxt.vif_c
			auto[FLUSH_EX]	100%	1 / 1 (100%)	48	1	state: coverpoint cnbxt.debug_cnbxt.vif_c
				100%	1 / 1 (100%)			state: coverpoint

Goal: universal UVM environment for CORE-V Verif



Features of “universal” core-v-verif




- Single UVM environment:
 - UVM Testcase has control of all components.
- **Corev-dv** for random instruction generator:
 - Wrapper for either **riscv-dv** or **FORCE-RISCV**.
 - “Generator Configuration” provides single test-specification interface to UVM testcases.
- Eliminate need for Toolchain:
 - Generators emit machine code directly.
 - Will maintain “legacy programmers interface” to Toolchain.
- **Corev-rm** for the core’s reference model:
 - Wrapper for either **Imperas OVPsim ISS** or **Spike** (others possible).
 - “RM Cfg” provides single test-specification interface to UVM testcases.
- Standardize “tracer” interfaces:
 - **RVFI** for cores.
 - **RVVI** for reference models.
- Functional Coverage model independent of Reference Model.
- Transaction Scoreboarding of retired instructions:
 - May retain step-and-compare functionality.
 - Eases support for out-of-order cores.
- New “Memory Agent” component:
 - Improved randomization of memory-to-core transactions.
 - Supports rich set of virtual peripherals.
 - Supports testcase constrainable memory interface parameters, PMA, etc.

Open Source HW Adoption Lessons

- Lesson 1 - Permissive use
 - permissive open-source licensing and processes to minimize business and legal risks
- Lesson 2 - IP quality
 - harness community best-in-class design and verification methods and contributions
- Lesson 3 - Roadmap & Ecosystem
 - ensure availability of IDE, RTOS / OS ports, physical design etc. and create a roadmap of cores covering a range of PPA metrics

OpenHW Cores Task Group



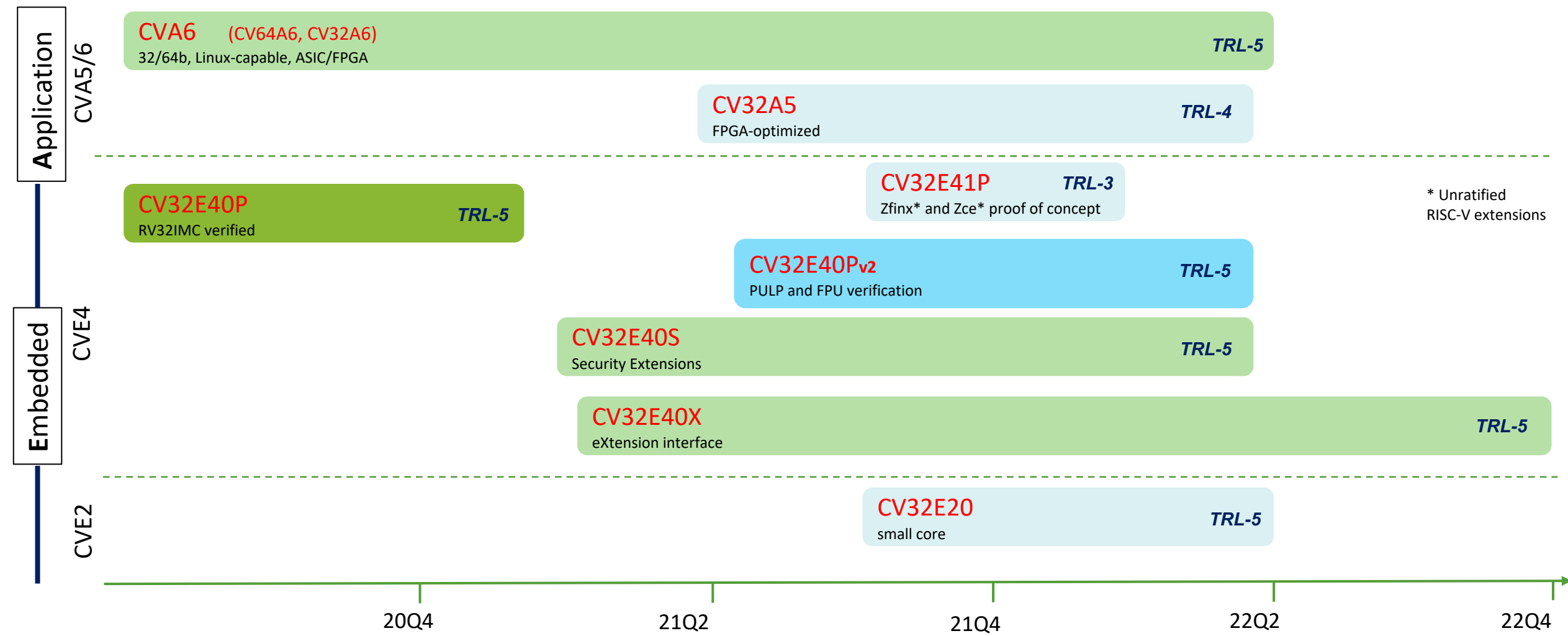
- Chair: Arjan Bink, Silicon Laboratories
- Vice-Chair: Jérôme Quévremont, Thales Research & Technology **THALES**
- develop feature and functionality roadmap and the open-source IP for the cores within the OpenHW Group such as the CORE-V Family of open-source RISC-V processors.
- Initial contribution of open-source RISC-V cores from [ETH Zurich PULP Platform](#) and the OpenHW Group is the [official committer for these repositories](#) **ETH zürich**  **PULP** Parallel Ultra Low Power

Core	Bits/Stage	Description
CVE4 (RISCY)	32bit / 4-stage	A family of 4-stage cores that implement, RV32IMFCXpulp, optional 32-bit FPU, instruction set extensions for DSP operations including HW loops, SIMD extensions, bit manipulation and post-increment instructions.
CVA6 (Ariane)	32 & 64bit / 6-stage	A family of 6-stage, single issue, in-order CPU cores implementing RV64GC extensions with three privilege levels M, S, U to fully support a Unix-like (Linux, BSD, etc.) operating system. The cores have configurable size, separate TLBs, a HW PTW and branch-prediction (branch target buffer, branch history table and a return address stack).





CORE-V™ Cores Roadmap



OpenHW SW Task Group



- Chair: Jeremy Bennett, Embecosm
- Vice-Chair: Yunhai Shang, Alibaba T-Head



- define, develop and support SW tool chain, operating system ports and firmware for the cores and IP developed within the OpenHW Group
- SW TG active projects include: GCC / LLVM, IDEs, FreeRTOS, HAL, CORE-V MCU SDK, etc.



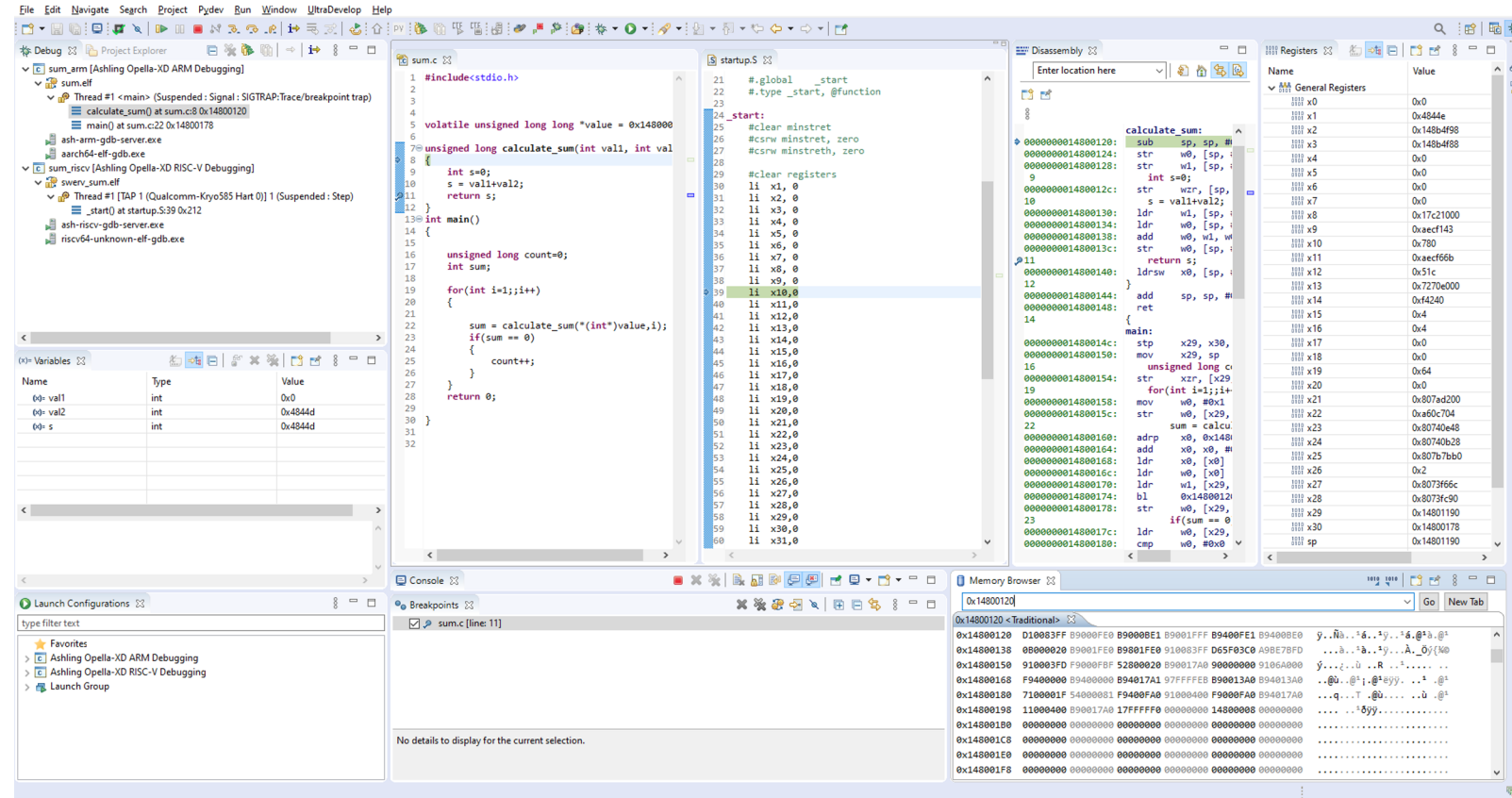


CORE-V™

IDE



- CORE-V IDE is an open-source development under the SW TG at the OpenHW Group
- Eclipse based IDE for CORE-V development
- Includes the GCC Toolchain for CORE-V
- OpenOCD Debug Support
- “Ready-to-run” examples for Digilent FPGA boards
- Getting started guides



OPENHW™
PROVEN PROCESSOR IP

© OpenHW Group

May 2022

31

OpenHW HW Task Group

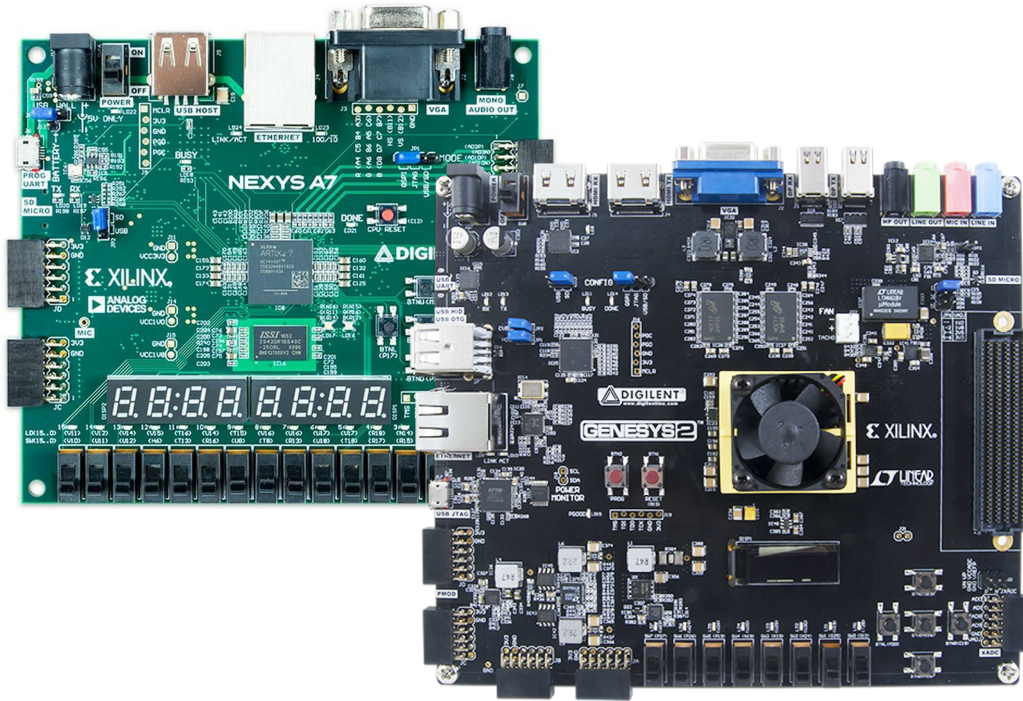
- Chair: Hugh Pollitt-Smith, CMC Microsystems
- Vice-Chair: Tim Saxe, QuickLogic
- define, develop and support SoC and FPGA based evaluation / development platforms for the cores and IP developed within the OpenHW Group.



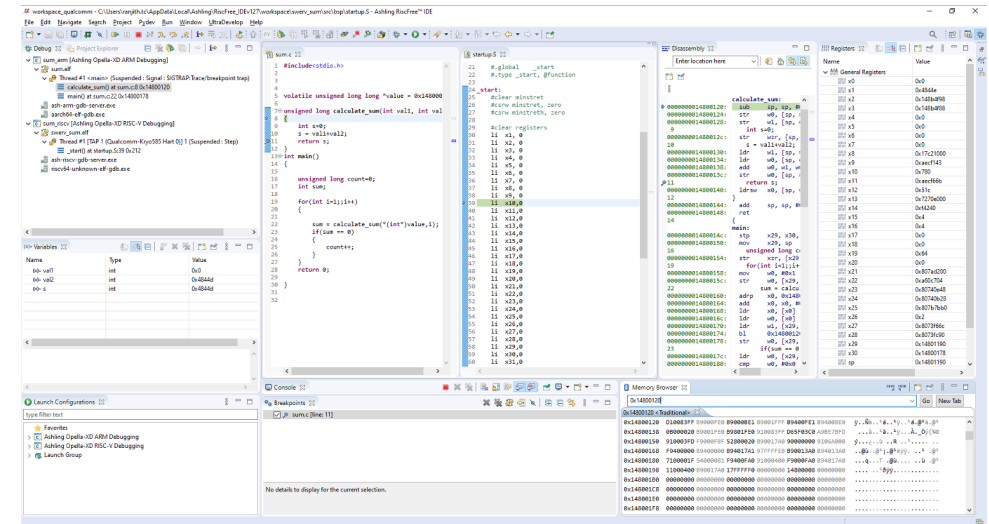


CORE-V™ FPGA Emulation

- CORE-V projects leverage Digilent NexysA7 & Genesys2 FPGA boards for soft-core bring up for both CVE4 and CVA6 Families



CORE-V™ IDE



ASHLING

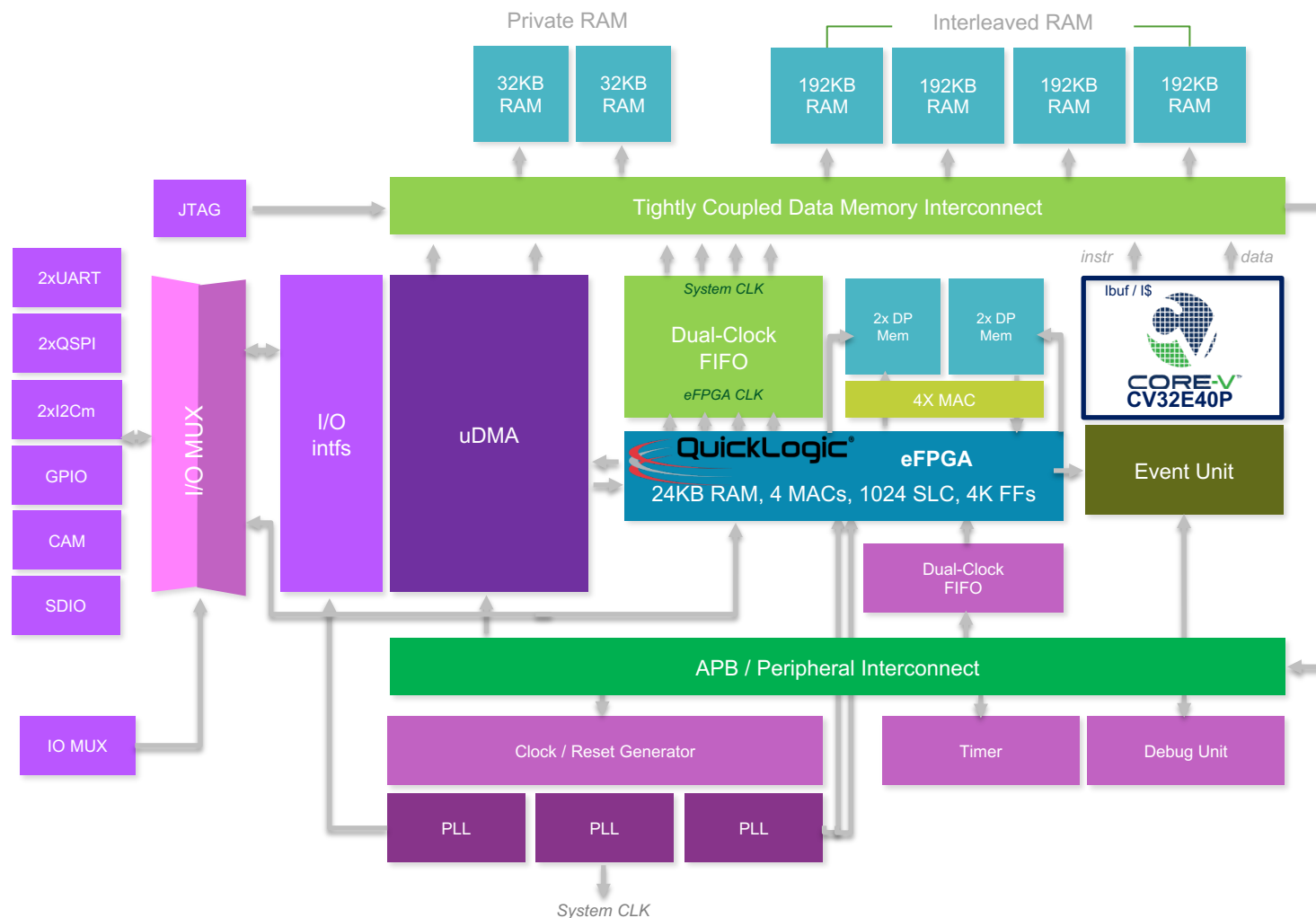
Opella LD
Debug Probe




GROUP™
OPENHW™
— PROVEN PROCESSOR IP —



CORE-V™ MCU Tapeout 2nd half 2022



- Real Time Operating System (e.g. FreeRTOS) capable ~600+MHz CV32E4 MCU
- Embedded FPGA fabric with hardware accelerators from QuickLogic
- Multiple low power peripheral interfaces (SPI, GPIO, I2C, HyperRAM, CAMIF, etc) for interfacing with sensors, displays, and connectivity modules
- Built in 22FDX at  GF®



Open HW ~2 Years Later Lessons Learned

- Lesson 1 - Permissive use
 - permissive open-source licensing and processes to minimize business and legal risks
- Lesson 2 - IP quality
 - harness community best-in-class design and verification methods and contributions
- Lesson 3 - Roadmap & Ecosystem
 - ensure availability of IDE, RTOS / OS ports, physical design etc. and create a roadmap of cores covering a range of PPA metrics



OPENHW^{GROUP}TM
— PROVEN PROCESSOR IP —

and



CORE-VTM



- OpenHW Group & CORE-V Family of open-source RISC-V cores for use in high-volume production SoCs
 - Visit www.openhwgroup.org for details
 - Visit OpenHW GitHub <https://github.com/openhwgroup> for projects
 - Learn more at [OpenHW TV](#)
- Follow us on Twitter [@openhwgroup](#) & [LinkedIn OpenHW Group](#)



OPENHW^{GROUP}TM
— PROVEN PROCESSOR IP —